

### **REMARKS**

Claims 1-56 were presented for examination. Claims 1-56 stand rejected. Thus, claims 1-56 are presently pending in this application, of which claims 1, 11, 21, 29, 39, and 49 are independent. Applicants submit that claims 1-56 are in condition for allowance.

The following comments address all stated grounds of rejection. Applicants respectfully urge the Examiner to pass the claims to allowance in view of the remarks set forth below.

### **Double Patenting Rejection**

#### **I. Claims 1-20 and 29-48 Provisionally Rejected Under Obviousness-Type Double Patenting Doctrine**

Claims 1, 11, 29, 39 are provisionally rejected under the judicially created doctrine of obviousness-type double patenting as unpatentable over claim 16 of co-pending U.S. Patent Application No. 10/190,288 ("Co-Pending Application"). Dependent claims 2-10, 12-20, 30-38, and 40-48 are also rejected only because they depend on rejected independent claims 1, 11, 29, and 39 respectively. Applicants traverse this rejection. Nevertheless, to obviate this rejection, Applicants submit herewith a terminal disclaimer with respect to this current application.

### **Claim Rejections Under 35 USC §103**

#### **II. Claims 1-20 and 29-48 Stand Rejected Under 35 U.S.C §103(a)**

Claims 1-20 and 29-48 stand rejected under 35 U.S.C. §103 as unpatentable over Shannon et al., "Mapping the Interface Description Language Type Model in C," November 1989, IEEE Transactions on Software Engineering, Vol. 15, No. 11 ("Shannon"), in view of Research Systems, "IDL", copyright 1994 ("Research Systems").

For ease of discussion, summaries of the claimed invention, Shannon, and Research Systems references are provided below.

A. Summary of Claimed Invention

The claimed invention is directed towards techniques for translating a call to a function in a first language into a call to the function in a second language. A definition of the function may be associated with a first language. The claimed invention processes the definition of the function to derive information about the function such that a call to the function can be provided in another language without also requiring the definition of the function. A call to a function includes an instruction or statement in a language to execute the function. The information is used to provide descriptive information about the function that may be stored in a library description file. The information may include the number of formal inputs and outputs to the function, the scope of the function, and the use of certain language features such as variable argument lists or the number of output arguments. Also, the information may identify a first interface for making a call to the function when the number of input and output arguments is known at translation time, and a second interface for making a call to the function when the number of input and output arguments is unknown at the time of translation.

The claimed invention uses the library description file to translate a call to a function in the first language into a call to the function in a second language without using the definition of the function provided in the first language. During translation of the first language into the second language, each time the translator processes a call to a function, the translator uses the information from the library description file to translate the call to the function into a corresponding call to the function in the second language. For example, if the translator

processes a line of MATLAB® source code having a function call of “F2(3)”, the call to the function F2 may be translated to a line of C source code that provides a call to the function “mlF2(mlfscalar(3.0), NULL, NULL);”. Instead of deriving the information from the definition of the function each time a call to the function is translated, the techniques of the claimed invention provide a faster process of translating a call to a function in one language to another language.

A. Summary of Shannon

Shannon discusses an interface description language (IDL) standard referred to as Diana. (See Shannon, at 1333). Diana is an intermediate representation of Ada programs that is used within Ada compilers. (See Shannon, at 1333). A translator tool maps descriptions representative of Ada into code fragments written in the target programming language. (See Shannon, at 1333). Shannon focuses on using an IDL, like Diana, to map the IDL structure specification into data structures in the C programming language. (See Shannon, at 1333).

B. Summary of Research Systems

Research Systems provides a scientific computing environment and programming language in which a scientist or scientific application developer may prototype and develop an application. (See Research Systems, page 1). Research Systems’ IDL combines mathematics, advanced data visualization, scientific graphics, and a graphical user interface toolkit to analyze and visualize scientific data. (See Research Systems, page 1). The IDL is a complete structured language and provides or supports math functions, 2D plotting, surface plotting, 3D graphics, mapping functions, image processing, general graphics, graphic output devices, GUI toolkits,

signal processing, statistics, and various input/output formats. (See Research Systems, pages 1-5). The IDL language of Research Systems is not the same IDL of an Interface Description Language as referred to in Shannon.

**B. Claims 1 and 29 Patentably Distinguished**

Independent claims 1 and 29 are directed towards a method and computer program product, respectively. These independent claims recite processing a definition of a function associated with a first language to create description information about the function. The description information being sufficient to enable *translation of a call to the function into a call to a corresponding function in a second language* without requiring processing of the definition of the function.

Shannon does not teach or suggest description information being sufficient to enable *translation of a call to the function into a call to a corresponding function in a second language* without requiring processing of the definition of the function. Shannon does not translate a call to a function, but rather, Shannon discusses mapping a data structure defined in an Interface Description Language (IDL) into a data structure declaration of a particular target language. That is, Shannon maps the definition of a data structure in the IDL into a definition of a corresponding data structure in a target language. In contrast to the claimed invention, Shannon does not discuss the *translation of a call to a function in a first language into a corresponding call to the function in a second language*. A call to a function is an instruction or statement to execute the function. Shannon is not concerned with translating a call to a function between languages but rather is concerned with mapping a definition of a data structure from IDL to a target language. As such, Shannon discusses the translation of data structures for a definition of

a function instead of translating a call to a function that will execute the function. Thus, Shannon fails to teach or suggest description information being sufficient to enable *translation of a call to the function into a call to a corresponding function in a second language* without requiring processing of the definition of the function.

In the Office Action, the Examiner indicates that Shannon does not disclose processing a definition of a function associated with the first language to create the description information. The Examiner cites Research Systems for the purpose of suggesting that one ordinarily skilled in the art might modify the IDL of Shannon to incorporate the IDL of Research Systems to process a definition of a function associated with the first language to create the description information. First, the IDL programming language of Research Systems is not analogous to the IDL or interface description language of Shannon. Rather, the IDL of Research Systems provides a specific computing environment and programming language instead of an interface description language. Furthermore, all the claim limitations must be taught or suggested by the prior art. As with Shannon, the programming language or computing environment of Research Systems does not teach or suggest description information being sufficient to enable *translation of a call to the function into a call to a corresponding function in a second language* without requiring processing of the definition of the function. The description of the computing environment and programming language of IDL of Research Systems lacks any discussions related to function call translation of the claimed invention. Thus, Research Systems fails to bridge the factual deficiencies of the Shannon reference.

For at least the aforementioned reasons, Shannon in view of Research Systems fails to detract from the patentability of claims 1 and 29. Claims 2-10 depend on and incorporate the patentable subject matter of independent claim 1. Claims 30-38 depend on and incorporate the

patentable subject matter of independent claim 29. As such, Shannon in view of Research Systems also fails to detract from the patentability of claims 2-10 and 30-38. Accordingly, Applicants respectfully request the Examiner to reconsider and withdraw the rejection of claims 1-10 and 29-38 under U.S.C §103.

C. Claims 11 and 39 Patentably Distinguished

Independent claims 11 and 49 are directed towards a method and computer program product respectively. These independent claims, as amended, recite providing a file of description items and using the file of description items to translate a first program file into a second program file. Each item of the description items includes description information about a function defined by a first language. The description information being sufficient to enable *translation of a call to the function into a call to a corresponding function in a second language* without requiring processing of the definition of the function.

Shannon does not teach or suggest description information being sufficient to enable *translation of a call to the function into a call to a corresponding function in a second language* without requiring processing of the definition of the function. Rather, Shannon discusses the mapping of a data structure definition from an Interface Description Language (IDL) to a particular target language. That is, Shannon provides a definition of a data structure in the target language based on the definition of the data structure in the IDL. In contrast to the claimed invention, Shannon does not discuss the *translation of a call to a function* in a first language *into a corresponding call to the function in a second language*. Instead of discussing function call translations between a first language and a second language, Shannon focuses on mapping definitions of data structures between IDL and a target language. As such, Shannon fails to teach

or suggest description information being sufficient to enable *translation of a call to the function into a call to a corresponding function in a second language* without requiring processing of the definition of the function.

In the Office Action, the Examiner indicates that Shannon does not disclose processing a definition of a function associated with the first language to create the description information. The Examiner cites Research Systems for the purpose of suggesting that one ordinarily skilled in the art might modify the IDL of Shannon to incorporate the IDL of Research Systems to process a definition of a function provided by the first language to create the description information. The IDL programming language of Research Systems is not an Interface Description Language (IDL) of Shannon. Although the IDL programming language can be used to build applications, the programming language of Research Systems does not teach or suggest description information being sufficient to enable *translation of a call to the function into a call to a corresponding function in a second language* without requiring processing of the definition of the function. Additionally, the IDL programming language does not teach or suggest processing a definition of a function associated with the first language to create the description information. The description of IDL of Research Systems does not provide any discussions related to a function call translation or processing a definition of a function of a language to create the description information. Thus, Research Systems fails to bridge the factual deficiencies of the Shannon reference.

For at least the aforementioned reasons, Shannon in view of Research Systems fails to detract from the patentability of claims 11 and 39. Claims 12-20 depend on and incorporate the patentable subject matter of independent claim 1. Claims 40-48 depend on and incorporate the patentable subject matter of independent claim 29. As such, Shannon in view of Research

Systems also fails to detract from the patentability of claims 12-20 and 40-48. Accordingly, Applicants respectfully request the Examiner to reconsider and withdraw the rejection of claims 11-20 and 39-48 under U.S.C §103.

### III. Claims 21-28 and 49-56 Stand Rejected Under 35 U.S.C §103(a)

Claims 21-28 and 49-56 stand rejected under 35 U.S.C. §103 as unpatentable over Elmroth et al., “A Web Computing Environment for the SLICOT Library,” December 2000, Brite-Euram III, Networks Programme NICONET (“Elmroth”) in view of Research Systems and in further view of Shannon.

For ease of discussions, a summary of the Elmroth reference is provided below.

#### A. Summary of Elmroth

Elmroth presents a prototype web computing environment for computations related to the design and analysis of control systems using the Subroutine Library In Systems and Control Theory (SLICOT) software library. (See Elmroth, Abstract and page 8). In this environment, a user supplies input data to the SLICOT routine that will be executed. (See Elmroth, Abstract and page 7). The routine then computes output data, the user receives the output data, and the data is stored on a server. (See Elmroth, Abstract and page 7). The environment enables the testing and use of SLICOT routines, which may be executed on a remote web server with no need for software installation on a local computer, allowing a user to test the routines with little effort. (See Elmroth, Abstract and page 8).



B. Claims 21 and 49 Patentably Distinguished

Independent claims 21 and 49 are directed towards a method and computer program product respectively. These independent claims, as amended, recite providing *a library file including functions defined by a first language*, and *processing the library file to create a function library* and a description file. *The function library includes one or more functions defined by a second language* and each function in the function library being a translated version of a function in the library file. The description file includes description information about each function in the library file. The description information being sufficient to enable *translation of a call to the function into a call to a corresponding function in the second language* without requiring processing of the definition of the function. These independent claims further recite using the description file to translate a program file from the first language into the second language, wherein *each call in the program file to a function in the library file is translated into a call to a corresponding function* in the second language.

Elmroth does not teach or suggest processing a library file including functions defined by a first language into a function library having translated versions of the functions in a second language. Rather, Elmroth provides access to a library via a web computing environment. Instead of converting functions from a library file in a first language to a function library of a second language, Elmroth provides for downloading matrix data in different formats, such as Fortran or MATLAB, on a local computer. As such, Elmroth is focused on data conversion and not function translation as in the claimed invention.

In the Office Action, the Examiner cites in Elmroth the interface between the Slicot Routines and the web interface and related scripts as disclosing one or more functions translated from the library file into a second language. Applicants respectfully disagree with the Examiner

and contend that Elmroth does not discuss processing a library file to have translated versions of functions in a function library. The interface cited by the Examiner provides a calling interface between the routines of the Slicot library presented on a web page to the web server environment for processing the web page input. The web interface and script libraries are not translated versions of the functions of the Slicot library. Furthermore, the cited interface of Elmroth does not process the Slicot library routines to provide translated versions in a function library.

Instead, a Slicot library routine calls an interface of the web interface to process data provided by the routine. The web interface may utilize scripts to process the data such as converting the data from one format to another format. As such, Elmroth fails to teach or suggest processing a library file including functions defined by a first language into a function library having translated versions of the functions in a second language.

Moreover, Elmroth as indicated by the Examiner does not teach or suggest description information being sufficient to enable *translation of a call to the function into a call to a corresponding function in a second language* without requiring processing of the definition of the function. The Examiner cites Shannon and Research Systems for the purpose of suggesting that one ordinarily skilled in the art might modify Elmroth to provide description information being sufficient to enable *translation of a call to the function into a call to a corresponding function in a second language* without requiring processing of the definition of the function. For the reasons discussed above in connection with the rejection of independent claims 1, 11, 21, and 39, Shannon and Research Systems fail to teach or suggest description information being sufficient to enable *translation of a call to the function into a call to a corresponding function in a second language* without requiring processing of the definition of the function. Therefore, Shannon and Research Systems fail to bridge the factual deficiencies of the Elmroth reference.

For at least the aforementioned reasons, Elmroth in view of Shannon in further view of Research Systems fails to detract from the patentability of claims 21 and 49. Claims 22-28 depend on and incorporate the patentable subject matter of independent claim 21. Claims 50-56 depend on and incorporate the patentable subject matter of independent claim 49. As such, Elmroth in view of Shannon in further view of Research Systems also fails to detract from the patentability of claims 22-28 and 50-56. Accordingly, Applicants respectfully request the Examiner to reconsider and withdraw the rejection of claims 21-28 and 49-56 under U.S.C §103.

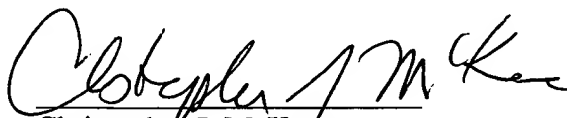
### **CONCLUSION**

In view of the remarks set forth above, Applicants contend that the presently pending claims in this application are patentable and in condition for allowance. Therefore, Applicants respectfully urge the Examiner to pass the claims to allowance.

If the Examiner deems there are any remaining issues, we invite the Examiner to call the Applicants' Attorney at the telephone number identified below.

Respectfully submitted,

LAHIVE & COCKFIELD, LLP



Christopher J. McKenna  
Registration No. 53,302  
Attorney For Applicants  
Lahive & Cockfield, LLP  
28 State Street  
Boston, Massachusetts 02109  
(617) 227-7400  
(617) 742-4214 (Fax)

Dated: June 24, 2005